<center>**REMARKS**</center>

Claims 1, 8, 14, 21, 27, and 34 have been amended. No claims have been added or canceled. Therefore, claims 1-39 remain pending in the application. Reconsideration is respectfully requested in view of the following remarks.

Claims Objected To But Otherwise Allowable:

Claims 9, 10, 22, 23, 35 and 36 were objected to as being dependent from a rejected base claim, but would be allowable if rewritten in independent form. Claims 11, 24 and 37 should also be indicated as allowable since they depended from allowable claims 9, 22 and 35, respectively.

Section 102(e) Rejection:

The Examiner rejected claims 1, 3, 14, 16, 27 and 29 under 35 U.S.C. § 102(e) as being anticipated by Gerard et al. (U.S. Patent 6,339,782) (hereinafter "Gerard"). Applicant respectfully traverses this rejection for at least the following reasons.

Applicant asserts that the Examiner's rejection is not fully and clearly stated. Applicant respectfully refers the Examiner to MPEP 707.07(d), which requires that, in an Examiner's Action, the ground of rejection should be "fully and clearly stated." Applicant also refers the Examiner to 37 CFR 1.104(c)(2) which requires that the pertinence of each reference "must be clearly explained." More specifically, while the Examiner cites various portions of the Gerard reference (e.g., column 9, lines 20-32; column 7, lines 17-20; and column 10, lines 55-61), the Examiner fails to clearly indicate which elements of Gerard that she considers to be equivalent to specific elements of Applicant's claim. For example, the Examiner fails to clearly indicate which elements of Gerard that she considers to be equivalent to Applicant's claimed one or more classes, persistence structure, and one or more enhanced classes. Accordingly, Applicant's assert that the Examiner has not stated a *prima facie* rejection.

Furthermore, as discussed below, Gerard fails to disclose numerous limitations of Applicant's claims.

In regard to claim 1, Gerard fails to teach or suggest a class structure based data object enhancer configured to <u>analyze the structure of the one or more classes to determine a persistence structure specifying data fields of the one or more classes to be persisted.</u> The Examiner cites column 9, lines 20-32 of Gerard, which is reproduced below:

> Another desirable change is to modify the new( ) java operator and its corresponding new( ) bytecode, which create new object instances of the specified class, to include parameters which specify whether or not the new object is to be persistent. If the new object is to be persistent, new( ) operator should specify the StoragePlugin object that is to create the persistent object. When compiled and run by the JVM, the new( ) bytecode should call the newObject(javaclass) method on the StoragePlugin object, with the class of the new object specified as a parameter. The StoragePlugin object then creates the specified new object. If the new object is not to be persistent, the new( ) bytecode creates regular transient object of the specified class.

Gerard teaches modifying Java's new( ) operator as well as its corresponding bytecode to include parameters which specify whether a new object (e.g., the object being instantiated) is to be persistent. If the object is to be persistent, the new( ) bytecode should call the newObject(javaclass) method in order to create a new object of the specified class (e.g., "javaclass"). If the object is not to be persistent, the new( ) bytecode creates a regular transient object. Nothing about Gerard's modifications of Java's new ( ) operator and corresponding bytecode teaches *analyzing the structure of classes*, much less analyzing the structure of one or more classes *to determine a persistence structure specifying data fields of the one or more classes to be persisted.*

Furthermore, Gerard fails to teach or suggest a class structure based data object enhancer configured to generate one or more enhanced classes corresponding to the one or more classes such that an object of the one or more enhanced classes is enhanced to persist data of the data fields to be persisted according to the

persistence structure, wherein said data of the data fields to be persisted is data of said object. The Examiner cites column 7, lines 17-20 and column 10, lines 55-61, which are reproduced below:

> When the newObject(javaclass) command on the StoragePlugin object 222 is called it creates a new persistent object of the specified class. This suitably would involve the allocation of storage both inside the JVM 200 and inside the persistent storage system 240. In particular, a referencing object is stored in JVM 200 that points to the real persistent object in persistent storage system 240. Likewise, when the destroyObject(ObjectID) command on the StoragePlugin object 222 is called it destroys the specified persistent object and frees the memory and/or other storage in which the object was kept. (column 7, lines 17-28).

> Turning to FIG. 4, a method 500 for creating and operating with persistent Java objects in accordance with the preferred embodiment is illustrated. The first step 502 is to provide a StoragePlugin object corresponding to a particular persistent storage system. As discussed above the StoragePlugin object is preferably implemented from a sub-class of objects corresponding to the type of persistent storage system and having implemented the commands described above. The next step 504 is to create a new persistent object using the newObject(javaclass) command on the StoragePlugin object. This command creates the object of the specified class and stores it in the persistent storage system. Because the persistent object can be of any class and does not require any persistent-object mixin class, this object persistence is orthogonal to the class of the object. (column 10, lines 47-61)

Gerard teaches objects of a StoragePlugin class that correspond to a particular persistent storage system. Such class includes commands [e.g., newObject(javaclass)] that are used for creating persistent objects of a specified class (e.g., javaclass). However, Gerard fails to mention anything at all about generating one or more enhanced classes corresponding to the one or more classes such that an object of the one or more enhanced classes is enhanced to persist data of the data fields to be persisted according to the persistence structure, wherein said data of the data fields to be persisted is data of said object. Presumably, the Examiner considers the StoragePlugin class to be equivalent to Applicant's claimed one or more enhanced classes; **however, Gerard does not teach that data of an object *of the StoragePlugin class* is persisted, much less persisted according to a persistence structure that specifies data fields to be persisted**.

Applicant reminds the Examiner that anticipation requires the presence in a single prior art reference disclosure of <u>each and every limitation</u> of the claimed invention, <u>arranged as in the claim</u>. M.P.E.P 2131; *Lindemann Maschinenfabrik GmbH v. American Hoist & Derrick Co.*, 221 USPQ 481, 485 (Fed. Cir. 1984). The **identical invention must** be shown <u>in as complete detail</u> as is contained in the claims. *Richardson v. Suzuki Motor Co.*, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989). As demonstrated above, Gerard clearly fails to teach or suggest several specific limitations of claim 1. Therefore, Gerard cannot be said to anticipate claim 1.

Thus, for at least the reasons presented above, Applicant's assert the rejection of claim 1 is unsupported by the cited art and removal thereof is respectfully requested. Similar remarks apply to independent claims 14 and 27.

Section 103(a) Rejections:

The Examiner rejected claims 2, 4-8, 15, 17-21, 28, 30-34, 11, 13, 24, 26, 37 and 39 under 35 U.S.C. § 103(a) as being unpatentable over Gerard in view of Calusinski (U.S. Publication 2005/0071342), and claims 12, 25 and 38 as being unpatentable over Gerard in view of Chan et al. (U.S. Patent 6,470,494) (hereinafter "Chan"). Applicant respectfully traverses these rejections for at least the reasons given above in regard to Gerard.

In regard to the § 102 and § 103 rejections, Applicants assert that numerous ones of the dependent claims recite further distinctions over the cited art. However, since the rejection has been shown to be unsupported for the independent claims, a further discussion of the dependent claims is not necessary at this time. Applicants reserve the right to present additional arguments.

CONCLUSION

Applicant submits that the application is in condition for allowance, and notice to that effect is respectfully requested.

If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5681-72300/RCK.

Respectfully submitted,

/Robert C. Kowert/
Robert C. Kowert, Reg. #39,255
Attorney for Applicant

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8850

Date: ___January 4, 2008___